

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

Claims 1-20 (canceled).

Claim 21 (currently amended): A method for generating a tuple representing relationships among connectors in a network, the method comprising:

identifying a link directly coupling a host to a first port of a first connector;

identifying an intermediate connection which indirectly couples the host to an intermediate port of an intermediate connector, wherein the intermediate connector is not aware of a connection to the first connector and the first connector is not aware of a connection to the intermediate connector; and

generating a new tuple identifying a relationship between the first connector and the intermediate connector based on the identified intermediate connection and the direct link between the host and the first port of the first connector, wherein the new tuple indicates that the first connector is directly coupled to the intermediate connector.

Claim 22 (previously presented): The method of claim 21, wherein the host is a singly-heard-host which is the only host heard on the first port of the first connector.

Claim 23 (previously presented): The method of claim 22, wherein the singly-heard-host is at least one of a workstation, a personal computer, a terminal and a printer.

Claim 24 (currently amended): The method of claim 21, wherein the tuple and the new tuple contain data associated with a-an updated topology of the network.

Claim 25 (previously presented): The method of claim 21, wherein generating the new tuple comprises:

determining that the first connector is directly coupled to the intermediate connector via a second port of the first connector.

Claim 26 (previously presented): The method of claim 25, further comprising:

storing the new tuple in the intermediate connector.

Claim 27 (previously presented): The method of claim 25, further comprising:

storing the new tuple in the first connector.

Claim 28 (previously presented): The method of claim 21, further comprising:

determining whether the host is heard only by the first port of the first connector;

if the host is heard only by the first port of the first connector, classifying the new tuple as a singly-heard host link tuple.

Claim 29 (previously presented): The method of claim 28, further comprising:

determining if another connector hears the host as a singly-heard host; and

if another connector hears the host, classifying the new tuple as a singly-heard conflict link tuple; and

resolving a conflict associated with the host between the first connector and the another connector.

Claim 30 (previously presented): The method of claim 28, further comprising:

generating an extra host link tuple for the intermediate connector indirectly coupled to the host via the intermediate connection.

Claim 31 (previously presented): The method of claim 30, further comprising:

examining the singly heard host link tuple and the extra host link tuple; and

based on the examining if the first connector is determined to be connected to the host and the intermediate connector is determined to be connected to the host, generating a conn-to-conn link tuple between the first connector and the intermediate connector.

Claim 32 (currently amended): A system for generating a tuple representing relationships among connectors in a network, the system comprising:

a first connector directly coupled by a direct link to a host via a first port;

an intermediate connector indirectly coupled to the host via an intermediate connection, wherein the intermediate connector is not aware of a connection to the first connector and the first connector is not aware of a connection to the intermediate connector; and

a tuple manager to generate a new tuple identifying a relationship between the first connector and the intermediate connector based on the intermediate connection and the direct link between the host and the first connector, wherein the new tuple indicates that the first connector is directly coupled to the intermediate connector.

Claim 33 (previously presented): The system of claim 32, wherein the host is a singly-heard-host which is the only host heard on the first port of the first connector.

Claim 34 (previously presented): The system of claim 33, wherein the singly-heard-host is at least one of a workstation, a personal computer, a terminal and a printer.

Claim 35 (currently amended): The system of claim 32, wherein the tuple and the new tuple contain data associated with a-an updated topology of the network.

Claim 36 (previously presented): The system of claim 32, wherein the tuple manager is to determine whether the first connector is directly coupled to the intermediate connector via a second port of the first connector.

Claim 37 (previously presented): The system of claim 32, further comprising:

a database to store the new tuple generated.

Claim 38 (previously presented): The system of claim 32, wherein the tuple manager is to further determine whether the host is heard only by the first port of the first connector and if the host is heard only by the first port of the first connector, the tuple manager is to classify the new tuple as a singly-heard host link tuple.

Claim 39 (previously presented): The system of claim 38, wherein the tuple manager is to further determine if another connector hears the host as a singly-heard host, classify the new tuple as a singly-heard conflict link tuple if another connector hears the host and resolve a conflict associated with the host between the first connector and the another connector.

Claim 40 (previously presented): The system of claim 32, wherein the tuple manager is to further generate an extra host link tuple for the intermediate connector indirectly coupled to the host via the intermediate connection, examine the singly heard host link tuple and the extra host link tuple and generate a conn-to-conn link tuple between the first connector and the intermediate connector if the first connector is determined to be connected to the host and the intermediate connector is determined to be connected to the host.

Claim 41 (new): A method for generating a tuple representing relationships among connectors in a network, the method comprising:

identifying a link directly coupling a host to a first port of a first connector;

identifying an intermediate connection which indirectly couples the host to an intermediate port of an intermediate connector, wherein the intermediate connector is not

aware of a connection to the first connector and the first connector is not aware of a connection to the intermediate connector;

generating a new tuple identifying a relationship between the first connector and the intermediate connector based on the identified intermediate connection and the direct link between the host and the first port of the first connector, wherein the new tuple indicates that the first connector is directly coupled to the intermediate connector;

determining whether the host is heard only by the first port of the first connector;

if the host is heard only by the first port of the first connector, classifying the new tuple as a singly-heard host link tuple;

determining if another connector hears the host as a singly-heard host;

if another connector hears the host, classifying the new tuple as a singly-heard conflict link tuple; and

resolving a conflict associated with the host between the first connector and the another connector.

Claim 42 (new): The method of claim 41, further comprising:

generating an extra host link tuple for the intermediate connector indirectly coupled to the host via the intermediate connection.

Claim 43 (new): The method of claim 42, further comprising:

examining the singly heard host link tuple and the extra host link tuple; and

based on the examining if the first connector is determined to be connected to the host and the intermediate connector is determined to be connected to the host, generating a conn-to-conn link tuple between the first connector and the intermediate connector.

Claim 44 (new): A system for generating a tuple representing relationships among connectors in a network, the system comprising:

a first connector directly coupled to a host via a first port;

an intermediate connector indirectly coupled to the host via an intermediate connection, wherein the intermediate connector is not aware of a connection to the first connector and the first connector is not aware of a connection to the intermediate connector;

a tuple manager to generate a new tuple identifying a relationship between the first connector and the intermediate connector based on the intermediate connection and the direct

link between the host and the first port of the first connector, wherein the new tuple indicates that the first connector is directly coupled to the intermediate connector and wherein the tuple manager is to further determine whether the host is heard only by the first port of the first connector and if the host is heard only by the first port of the first connector, the tuple manager is to classify the new tuple as a singly-heard host link tuple and the tuple manager is to further determine if another connector hears the host as a singly-heard host, classify the new tuple as a singly-heard conflict link tuple if another connector hears the host and resolve a conflict associated with the host between the first connector and the another connector.

Claim 45 (new): The system of claim 44, wherein the tuple manager is to further generate an extra host link tuple for the intermediate connector indirectly coupled to the host via the intermediate connection, examine the singly heard host link tuple and the extra host link tuple and generate a conn-to-conn link tuple between the first connector and the intermediate connector if the first connector is determined to be connected to the host and the intermediate connector is determined to be connected to the host.